Sparse non-negative matrix factorization for retrieving genomes across metagenomes

Vincent Prost[1,2,3], Stéphane Gazut[1], and Thomas Brüls[2,3]

[1] CEA, LIST, Laboratoire Sciences des Données et de la Décision, 91191 Gif-sur-Yvette, France
[2] CEA, DRF, Institut Jacob, Genoscope, 91057 Evry, France
[3] CNRS-UMR8030, Université Paris-Saclay, UEVE, 91057 Evry, France
{vincent.prost, stephane.gazut, thomas.bruls}@cea.fr

Abstract. The development of massively parallel sequencing technologies enables to sequence DNA at high-throughput and low cost, fueling the rise of metagenomics which is the study of complex microbial communities sequenced in their natural environment. A metagenomic dataset consists of billions of unordered small fragments of genomes (reads), originating from hundreds or thousands of different organisms. The *de novo* reconstruction of individual genomes from metagenomes is practically challenging, both because of the complexity of the problem (sequence assembly is NP-hard) and the large data volumes. The clustering of sequences into biologically meaningful partitions (e.g. strains), known as binning, is a key step with most computational tools performing read assembly as a pre-processing. However, metagenome assembly (and even more cross-assembly) is computationally intensive, requiring terabytes of memory; it is also error-prone (yielding artefacts like chimeric contigs) and discards vast amounts of information in the form of unassembled reads (up to 50% for highly diverse metagenomes). Here we show how online learning methods for sparse non-negative matrix factorization can recover relative abundances of genomes across multiple metagenomes and support assembly-free read binning by using abundance covariation signals derived from the occurrence of unique k-mers (subsequences of size k) across samples. The combinatorial explosion of k-mers is controlled by indexing them using locality sensitive hashing, and sparse coding and dictionary learning techniques are used to decompose the k-mer abundance covariation signal into genome-resolved components in latent space.

Keywords: Non negative matrix factorization · Sparse coding · Dictionary learning · Online learning · Metagenomics · Clustering

1 Introduction

Metagenomics leverages high-throughput sequencing to extract genomic information about microbial communities *in situ*. Metagenomic studies have already expanded knowledge in various domains, like microbial ecology or human medicine through the study of human-associated microbiotas. Metagenomic datasets consist in billions of unordered small genome fragments, typically a few hundreds

base pairs (bp) at best, which is very small compared to the size of a typical bacterial genome (about 10^6bp). Sequences are randomly sampled, i.e. without knowing the genome nor the position in the genome they are derived from, often leading to highly fragmented assemblies.

Many computational approaches seek to solve an intermediate problem called binning, that aims at grouping together reads originating from the same genome. We can broadly distinguish two different binning strategies. The first one relies on *de novo* assembly [14] as a pre-processing step and performs binning at the contig level.The main advantage here is a reduction in the number of objects to be clustered and more robust compositional signals associated with longer sequences. Performing *de novo* assembly on metagenomic datasets is however computationally intensive, especially in terms of computer memory. It is also a source of artefacts and discards significant amounts of raw data.

A second strategy is to perform binning at the (unassembled) read-level, followed by targeted assemblies of the resulting lower complexity partitions (see subsection 2.2). A larger number of objects need to be dealt with upfront, but this approach has the potential to avoid important drawbacks of *de novo* assembly, like biases against low-abundance sequences. We describe here such a "bin-first assemble-second" method that achieves read-level binning by decomposing the k-mer abundance covariation signal into latent genomes using online sparse non-negative matrix factorization.

2 Related work

2.1 Clustering unique *k*-mers

Many binning methods (e.g. [15]) exploit occurrences of unique k-mers (substrings of size k). With sufficiently long k-mers (e.g. $k > 20$), we can assume that most of them will be genome-specific. Solving the binning problem is therefore equivalent to clustering unique k-mers. The Lander-Waterman model [9] assumes that random sequencing will lead to Poisson distributed nucleotide coverage, which provides a rationale for binning k-mers from a given mixture of genomes. Assuming that occurrences of unique k-mers are Poisson distributed with parameter λ_i proportional to the abundance of the genome it comes from, the count $n(w_j)$ of a k-mer w_j is Poisson distributed : $P(n(w_j) = c) = Poisson(\lambda_i; c)$ where $Poisson(\lambda_i; c)$ is the probability of a Poisson random variable taking the value c. The parameters λ_j can be estimated by maximum likelihood using an Expectation-Maximization (EM) algorithm[3], see for example [15]. Reads are then partitioned into bins according to their k-mer content and the estimated parameters.

2.2 Clustering *k*-mers across multiple samples

Abundance signals were originally used to cluster k-mers and reads within individual samples. To handle experimental setups involving multiple samples,

abundance covariation signals can be used to cluster reads from individual taxa across samples, see for example [11] which proceeds by analogy to LSI (Latent Semantic Indexing [5], a classical method for document classification), by projecting each sample into the singular vector space with SVD (singular value decomposition): $X = U\Sigma V^T$ where U and V are orthogonal and Σ is diagonal. k-mers can then be clustered by doing fixed radius k-means on the lines of V[11].

3 Material and methods

3.1 Indexing *k*-mers by locality sensitive hashing

As the number of possible k-mers can be very large (e.g. $k = 30$ gives $4^{30} \approx 10^{18}$ possible k-mers), the computer memory needed for storing the counts of all observed k-mers can become prohibitive. The use of inverted indexes as in [15] will not be tractable for large or multi-samples datasets. Ref [11] proposes to use Locality Sensitive Hashing (LSH), a technique initially used to improve nearest-neighbour searches in high dimensions [6].

Each k-mer w_i is represented in a k-dimensional complex vector space $\mathbb{C}^k$, for example by using a mapping for each letter of the form: $A = 1, C = i, G = -i, T = -1$. Those numbers can further be weighted by a quality score, which informs about base call confidence. d random hyperplanes are then drawn in $\mathbb{C}^k$, e.g. with normal vector $v_j \in \mathbb{C}^k$. Each hyperplane separates the space into two half spaces with the hashing function : $h_j(w_i) = sign(w_i^T v_j) \in \{-1, 1\}$, and therefore defining 2^d subspaces or buckets.

Thus, each k-mer w_i, initially living in a space of cardinal 4^k, can be associated to a binary code $(h_1(w_i), h_2(w_i), .., h_d(w_i)) \in \{-1, 1\}^d$ of size d and be represented in a space of cardinal 2^d. This way the size of the "dictionary" can be controlled via the number of hyperplanes selected.

3.2 Sparse non negative matrix factorization

The experimental k-mer count data can be viewed as a sparse composition of positive components, hence non negative matrix factorization (NMF) stands out as a natural analytical paradigm. This technique was originally proposed by Lee and Seung [10]. In NMF, the goal is to approximate the sample by k-mer count matrix $X \in \mathbb{R}^{n \times 2^d}$ by the product of two non-negative matrices $U \in \mathbb{R}^{n \times K}$ and $V \in \mathbb{R}^{2^d \times K}$, usually by finding a solution to the following constrained minimization problem:

$$U, V = \underset{U,V \geq 0}{\operatorname{argmin}} \mathcal{D}(X||UV^T) + J(U, V) \tag{1}$$

where $\mathcal{D}$ is an error function and J a penalty term ensuring sparsity or regularity on U and V. In our model, the values of X are sparse compositions of k-mer

counts. If we denote S_{isj} the count of a k-mer specific to genome s appearing in bucket j and sample i, then:

$$X_{ij} = \sum_{s=1}^{K} S_{isj} \tag{2}$$

S_{isj} follows a Poisson distribution of parameter $U_{is}V_{js}$, where U_{is} is proportional to the abundance of genome s in sample i and V_{js} is a sparse coefficient. Due to the additivity of the Poisson distribution, X_{ij} is also Poisson distributed:

$$P(X_{ij} = c) = Poisson\left(\sum_{s=1}^{K} U_{is}V_{js}; c\right) \tag{3}$$

As stated originally in [10] and elaborated in [2], maximizing the likelihood of this model is equivalent to solving (1) when $\mathcal{D}$ is the Kullback-Leibler divergence:

$$KL(X||UV^T) = \sum_i \sum_j \log \frac{X_{ij}}{[UV^T]_{ij}} - X_{ij} + [UV^T]_{ij} \tag{4}$$

Lee and Seung [10] proposed an iterative algorithm for solving eq. (1) when $J := 0$. We refer to this algorithm in the experiments as "L&S-KL".

We can expect V to be very sparse, and this sparsity depends on the expected number of k-mers sharing a same bucket. It therefore depends on the number of buckets 2^d compared to the number of distinct k-mers measured, the latter being related to the k-mer size selected and the genome diversity in the metagenomes analyzed. We therefore need to set constraints in the optimization problem to enforce the sparsity of V, by either penalizing the l_0 or l_1 norm of the lines of V: $J(U,V) = \beta \sum_{i=1}^{2^d} \|v_i\|_\gamma, \gamma \in \{0,1\}$, where v_i is the ith line of V. On the other hand, we will frequently face situations where $n < K$, hence resorting to sparsity constraints in order to find solutions with the fewest number of non zeros among the infinite number of solutions [1].

3.3 Online dictionary learning

Iterative methods like [10] require to keep the whole dataset in memory, which is not always tractable when the right matrix V has large dimensions. The convergence speed can also be slow. Following Mairal et al., [12], we use an online dictionary learning method that aims to solve eq. (1) when $\mathcal{D}(X||UV^T) = \|X - UV^T\|_2^2$, where $\|.\|_2$ is the Frobenius norm. For each new coming input x_t, it proceeds in an online fashion by alternating a sparse coding step (updating v_t) :

$$v_t = \underset{v \geq 0}{\operatorname{argmin}} \|x_t - Uv\|_2^2 + \lambda \|v\|_1 \tag{5}$$

and a dictionary update step (updating matrix U) :

$$U^{t+1} = \underset{U \in \mathcal{C}}{\operatorname{argmin}} \sum_{i=1}^{t} \|x_i - Uv_i\|_2^2 \tag{6}$$

where x_i denotes the ith column of X, v_i the ith line of V, and $\mathcal{C}$ is a convex set. We evaluated the method from Mairal and colleagues with different sparse coding steps, and refer to it as lasso-DL when the sparse coding step is a lasso regression like eq. (5) and omp-DL when it is Orthogonal Matching Pursuit, aiming to solve :

$$v_t = \underset{v \geq 0}{\operatorname{argmin}} \|v\|_0 \text{ subject to } \|x_t - Uv\|_2 < \epsilon \tag{7}$$

Ref [13] proposes an algorithm inspired by [12] but aiming to solve problem (1) when $D(.||.) := KL(.||.)$ and with sparsity constraints; it will be noted KL-DL in our experiments.

3.4 Data

We analyzed two types of datasets: i) synthetic datasets simulating k-mer counts in a sparse Poisson factor model, cf. eq. (2), ii) semi-synthetic datasets simulating the sequencing of microbial communities by randomly sampling sequences from controlled mixtures of real genomes.

The first dataset was used to evaluate the performance of the different algorithms in the ideal case of the Lander-Waterman model. We have control over the variables of the model and evaluate the ability of the methods to retrieve the underlying abundances. In the second dataset, real genomes are used to simulate a random shotgun sequencing experiment. In this case, we evaluate the final binning results by quantifying the ability of the different algorithms to cluster reads into genome-resolved partitions with precision and recall metrics (P/R).

4 Results

4.1 Synthetic data

Following [1], we first evaluate the online learning algorithms on synthetic signals, given random underlying abundance parameters $A \in \mathbb{R}^{n \times K}$. At each iteration, T samples $(x_1, x_2, .., x_T)$ are independently drawn following eq. (8)

$$x_i^{(j)} = \sum_{k=1}^{K} \pi_k s_k^{(j)}, \quad \pi_k \sim Binomial(p, 3), \quad s_k^{(j)} \sim Poisson(A_{j,k}) \tag{8}$$

where $x^{(j)}$ denotes the jth coordinate of vector x. The computed left matrix U was compared against the known abundance parameter A, with the error defined as the sum of quadratic errors between the columns of A and the closest columns of U : $error = \sum_{k=1}^{K} \min_i \frac{\|A_{.,k} - U_{.,i}\|^2}{\|A_{.,k}\| \|U_{.,i}\|}$. The tests were carried out with parameters $p = 0.05, K = 140, n = 20, T = 1000$. Fig. 1 shows the comparison of different online learning algorithms. The curve "k-means" represents the online version of k-means (sequential k-means [4]), while "kl-means" is the same algorithm but with euclidean distance replaced by KL-divergence. All methods with

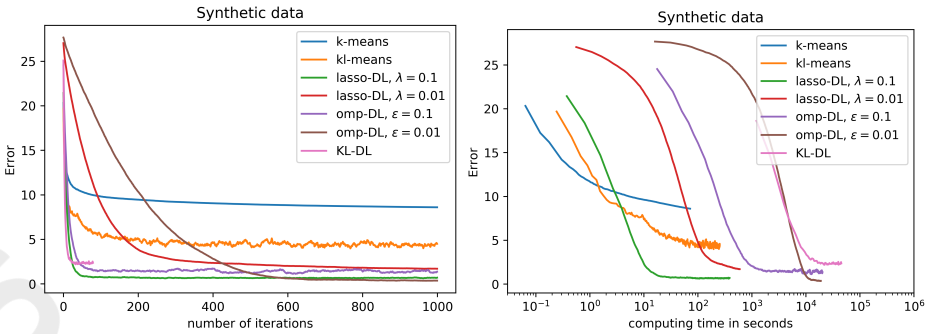

Fig. 1. Left: error as a function of iteration number. Right: error as a function of computation time (on a logarithmic scale).

relaxed sparse constraints performed better than k-means in recovering underlying abundances. Using an OMP sparse coding step tends to slightly improve the estimation at the expense of a higher computing cost. Lasso-DL achieves the best convergence speed both in iteration number and computation time. KL-DL is much slower than other methods and surprisingly fails to improve the final estimation.

4.2 Semi-synthetic data

We tested the algorithms on artificial metagenomic datasets simulating a cohort of $n = 50$ samples (individuals) [7]. Each sample contains a random subset (of size Genome nb) of (real) bacterial genomes that are randomly sampled to generate shotgun reads (see [7] for details). The number of reads varies from 200,000 (for Genome nb = 20) to 7,5 millions (for Genome nb = 700) per sample.

In the sequence binning process, k-mers are initially segregated, the matrix V is computed and k-mers of bucket i are assigned to cluster k if $k = \text{argmax}_j V_{i,j}$, as in [16]. Following this step, sequences are then assigned to the clusters based on their k-mer content, as in [11]. The sequence data was pre-processed by hashing (cf. subsection 3.1) with $d = 27$ (Table 1) and $d = 30$ (Table 2). The number of clusters K was the same for all methods and fixed to $1.5 \times$Genome nb. Parameters λ and ϵ were set so as to achieve a good compromise between the sparsity of V and the reconstruction error.

It can be seen that the canonical (non regularized) NMF algorithm of Lee and Seung[10] performs best when $n > K$, while the sparse NMF variants outperform the others in the under-determined regime, and that overall the performances decrease with the complexity of the genome mixtures (Table 1). The online dictionary learning methods scale well to large dimensions (Table 2) (neither L&S-KL nor KL-DL could be evaluated for $d = 30$ due to prohibitive computing times) and perform better than the state of the art pre-assembly binning algorithm LSA[11].

Table 1. Binning performance on semi-synthetic metagenomic datasets ($d = 27$).

Genome nb	k-means		L&S-KL		omp-DL		lasso-DL		KL-DL		LSA	
	P/R		P/R		P/R		P/R		P/R		P/R	
20	78.5	82.1	83.6	90.0	72.6	71.1	76.9	80.1	78.2	90.7	77.9	77.4
100	80.2	76.0	65.8	66.6	80.5	77.2	80.1	77.0	79.2	80.3	74.9	79.3
200	69.1	67.8	51.7	50.7	70.0	69.9	70.0	68.3	62.3	70.0	56.9	61.4
700	46.6	44.4	30.9	28.7	48.6	50.2	49.1	47.5	-	-	28.8	40.8

Table 2. Binning performance on semi-synthetic metagenomic datasets ($d = 30$).

Genome nb	omp-DL		lasso-DL		LSA	
	P/R		P/R		P/R	
100	93.0	92.1	95.2	92.9	87.3	90.1
200	86.4	90.8	87.2	90.4	70.0	88.4
700	75.2	80.7	76.1	83.3	50.9	78.5

5 Conclusion

We have shown that sparse non negative matrix factorization can be successfully applied to the analysis of metagenomic data. We explored and compared different methods and validated them through experiments on synthetic and semi-synthetic datasets. We have demonstrated that online dictionary learning methods coupled with sparse coding are able to recover underlying parameters in a sparse Poisson factor model and in an under-determined regime that is relevant for the analysis of real-world metagenomic data.

The online dictionary learning method was also applied to a massive real-world metagenomic dataset derived from human gut microbiota, comprising more than 10^{10} reads encompassing 10 terabytes of sequence data. Results from these analyses are described elsewhere [8], and illustrate the ability of the sparse coding method to scale to very large datasets and to recover low-abundance genomes that are typically missed by assembly-based approaches.

Acknowledgments This work was mainly funded by the office of the High Commissioner of CEA. The authors would like to thank Olexiy Kyrgyzov for sharing some computer code and for the analysis of the real dataset.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. IEEE Transactions on Signal Processing **54**(11), 4311–4322 (Nov 2006). https://doi.org/10.1109/TSP.2006.881199
2. Cemgil, A.T.: Bayesian inference for non-negative matrix factorisation models. Intell. Neuroscience **2009**, 4:1–4:17 (Jan 2009). https://doi.org/10.1155/2009/785152, `http://dx.doi.org/10.1155/2009/785152`

3. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society: Series B (Methodological) **39**(1), 1–22 (1977)
4. Duda, R.O.: Pattern recognition for hci. `www.cs.princeton.edu/courses/archive/fall08/cos436/Duda/PR_home.htm` (June 1997), accessed: 2019-05-27
5. Dumais, S.T.: Latent semantic analysis. Annual Review of Information Science and Technology **38**(1), 188–230 (2004). https://doi.org/10.1002/aris.1440380105, `https://onlinelibrary.wiley.com/doi/abs/10.1002/aris.1440380105`
6. Gionis, A., Indyk, P., Motwani, R.: Similarity search in high dimensions via hashing. In: Proceedings of the 25th International Conference on Very Large Data Bases. pp. 518–529. VLDB '99, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1999), `http://dl.acm.org/citation.cfm?id=645925.671516`
7. Gkanogiannis, A., Gazut, S., Salanoubat, M., Kanj, S., Brüls, T.: A scalable assembly-free variable selection algorithm for biomarker discovery from metagenomes. BMC bioinformatics **17**(1), 311 (2016)
8. Kyrgyzov, O., Prost, V., Gazut, S., Farcy, B., Brüls, T.: Binning unassembled short reads based on k-mer covariance using sparse coding. bioRxiv (2019). https://doi.org/10.1101/599332, `https://www.biorxiv.org/content/early/2019/05/07/599332`
9. Lander, E.S., Waterman, M.S.: Genomic mapping by fingerprinting random clones: A mathematical analysis. Genomics **2**(3), 231 – 239 (1988). https://doi.org/https://doi.org/10.1016/0888-7543(88)90007-9, `http://www.sciencedirect.com/science/article/pii/0888754388900079`
10. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization. In: Proceedings of the 13th International Conference on Neural Information Processing Systems. pp. 535–541. NIPS'00, MIT Press, Cambridge, MA, USA (2000), `http://dl.acm.org/citation.cfm?id=3008751.3008829`
11. Lowman Cleary, B., Lauren Brito, I., Huang, K., Gevers, D., Shea, T., Young, S., J Alm, E.: Detection of low-abundance bacterial strains in metagenomic datasets by eigengenome partitioning. Nature biotechnology **33** (09 2015). https://doi.org/10.1038/nbt.3329
12. Mairal, J., Bach, F., Ponce, J., Sapiro, G.: Online learning for matrix factorization and sparse coding. J. Mach. Learn. Res. **11**, 19–60 (Mar 2010), `http://dl.acm.org/citation.cfm?id=1756006.1756008`
13. Nguyen, D., Ho, T.: Fast parallel randomized algorithm for non-negative matrix factorization with kl divergence for large sparse datasets. International Journal of Machine Learning and Computing **6** (04 2016). https://doi.org/10.18178/ijmlc.2016.6.2.583
14. R Zerbino, D., Birney, E.: Velvet: Algorithms for de novo short read assembly using de bruijn graphs. Genome research **18**, 821–9 (06 2008). https://doi.org/10.1101/gr.074492.107
15. Wu YW, Y.Y.: A novel abundance-based algorithm for binning metagenomic sequences using l-tuples. J Comput Biol **18**(3), 523–34 (2011). https://doi.org/https://doi.org/10.1016/0888-7543(88)90007-9, `http://www.sciencedirect.com/science/article/pii/0888754388900079`
16. Xu, W., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 267–273. SIGIR '03, ACM, New York, NY, USA (2003). https://doi.org/10.1145/860435.860485, `http://doi.acm.org/10.1145/860435.860485`